We claim:

1. A security method comprising:

(a) digitally signing a first load module with a first digital signature designating the first load module for use by a first device class;

(b) digitally signing a second load module with a second digital signature different from the first digital signature, the second digital signature designating the second load module for use by a second device class having a tamper resistance and/or work factor substantially different from the tamper resistance and/or work factor of the first device class;

(c) distributing the first load module for use by at least one device in the first device class; and

(d) distributing the second load module for use by at least one device in the second device class.

2. A method as in claim 1 further including the step of using the first and second digital signatures to prevent the tamper

resistances and/or work factors of the first and second device classes to become equal.

3. A method as in claim 1 further including the step of conditionally executing, based at least in part on authenticating the first digital signature, the first load module with a first electronic appliance within the first device class.

4. A method as in claim 3 further including the step of conditionally executing, based at least in part on authenticating the second digital signature, the second load module with a second electronic appliance different from the first electronic appliance, the second electronic appliance being within the second device class.

5. A software verifying method comprising:

(a) testing a load module having at least one specification associated therewith;

(b) verifying that the load module satisfies the specification; and

(c) issuing at least one digital certificate attesting to the results of the verifying step.

6. A method of authenticating a load module comprising:

(a) authenticating a first digital signature associated with the load module, including the step of employing a first one-way hash algorithm, a first decryption algorithm, and a first public key; and

(b) authenticating a second digital signature associated with the load module, including the step of employing at least one of:

(i) a second one-way hash algorithm that is dissimilar to the first one-way hash algorithm,

(ii) a second decryption algorithm that is dissimilar to the first decryption algorithm, and

(iii) a second public key that is dissimilar to the first public key.

7. A method as in claim 6 further including the step of randomly selecting one of step (a) and step (b) prior to executing the load module.

8. A method as in claim 6 wherein:

(i) step (a) is performed by a first electronic appliance, and

(ii) step (b) is performed by a second electronic appliance different from the first electronic appliance.

9. A protected processing environment comprising:

means for providing a tamper resistance enclosure,

means for maintaining at least one public verification key within the tamper resistant enclosure, and

means for authenticating load modules based, at least in part, on use of the public verification key.

10. A method of distinguishing between trusted and untrusted load modules comprising:

(a) receiving a load module,

(b) determining whether the load module has an associated digital signature,

(c) if the load module has an associated digital signature, authenticating the digital signature using at least one secret public key; and

(d) conditionally executing the load module based at least in part on the results of authenticating step (c).

11. A method of increasing the security of a virtual distribution environment comprising plural interoperable protected processing environments having different work factors, the method comprising:

(a) classifying the plural protected processing environments based on work factor,

(b) distributing different verification public keys to different protected processing environments having different work factor classifications, and

(c) using the distributed verification public keys to authenticate load modules, including the step of preventing protected processing environments having different work factor classifications from executing the same load module.

12. A method as in claim 11 further including the step of maintaining the distributed verification public keys within tamper resistant enclosures.

13. A method as in claim 11 further including the step of digitally signing each load module with at least two substantially different, independent techniques.

14. A method as in claim 11 further including the step of testing whether the load module satisfies at least one specification, and digitally signing the load module and the associated specification if the testing step reveals the specification is satisfied.

15. A protected processing comprising:

a tamper resistant barrier having a first work factor, and

at least one arrangement within the tamper resistant barrier that prevents the protected processing environment from executing the same load module accessed by a further protected processing environment having a further tamper resistant barrier with a further work factor substantially different from the first work factor.

16. A protected processing environment as in claim 15 wherein the preventing arrangement includes a digital signature authenticating circuit.

17. A protected processing environment as in claim 15 wherein the preventing arrangement includes first and second digital signature authenticating circuits applying substantially different digital signature authenticating techniques.

18. A protected processing environment as in claim 15 wherein the preventing arrangement comprises means for randomly

selecting between first and second, substantially different digital signature authentication techniques.

19. A method for protecting a computation environment surrounded by a tamper resistant barrier having a first work factor, the method including:

preventing the computation environment from using the same software module accessible by a further computation environment having a further tamper resistant barrier with a further work factor substantially different from the first work factor.

20. A method as in claim 19 wherein the preventing step comprises authenticating at least one digital signature associated with the first-mentioned computation environment as corresponding to the first work factor.

21. A method of protecting computation environments comprising:

(a) associating plural digital signatures with a load module;

(b) authenticating a first subset of the plural digital signatures with a first tamper resistant computation environment; and

(c) authenticating a second subset of the plural digital signatures with a second tamper resistant computation environment different from the first environment.

22. A computer security method comprising:

digitally signing, using a first digital signing technique, a first executable designating the first executable for use by a first device class; and

digitally signing, using a second digital signing technique different from the first digital signing technique, a second executable designating the second executable for use by a second device class having a tamper resistance and/or work factor substantially different from the tamper resistance and/or work factor of the first device class.

23. A method as in claim 22 further including the step of using the first and second digital signatures to prevent the tamper resistances and/or work factors of the first and second device classes from collapsing into one another.

24. A method as in claim 22 further including the step of conditionally executing the first executable based at least in part on authenticating the first executable with a first electronic appliance within the first device class.

25. A method as in claim 24 further including the step of conditionally executing the second executable with a second electronic appliance different from the first electronic appliance, the second electronic appliance being within the second device class.

26 A software verifying method comprising:

testing a executable having at least one specification associated therewith;

verifying that the executable satisfies the specification; and

issuing at least one digital certificate attesting to the results of the verifying step.

27. A method of authenticating a executable comprising:

(a) authenticating a first digital signature associated with the executable, including the step of employing a first one-way hash algorithm, a first decryption algorithm, and a first public key; and

(b) authenticating a second digital signature associated with the executable, including the step of employing at least one of:

(i) a second one-way hash algorithm that is dissimilar to the first one-way hash algorithm,

(ii) a second decryption algorithm that is dissimilar to the first decryption algorithm, and

(iii) a second public key that is dissimilar to the first public key.

28. A method as in claim 27 further including the step of randomly selecting one of step (a) and step (b) prior to executing the executable.

29. A method as in claim 27 wherein:

(i) step (a) is performed by a first electronic appliance, and

(ii) step (b) is performed by a second electronic appliance different from the first electronic appliance.

30. A secure execution space comprising:

means for providing a tamper resistant barrier,

means for maintaining at least one public verification key within the tamper resistant barrier, and

means for authenticating executables based, at least in part, on use of the public verification key.

31. A method of distinguishing between trusted and untrusted executables comprising:

(a) receiving a executable,

(b) determining whether the executable has an associated digital signature,

(c) if the executable has an associated digital signature, authenticating the digital signature using at least one secret public key; and

(d) conditionally executing the executable based at least in part on the results of authenticating step (c).

32. A method of increasing the security plural interoperable secure execution spaces having different work factors, the method comprising:

(e) classifying the plural secure execution spaces based on work factor,

(f) distributing different verification public keys to different secure execution spaces having different work factor classifications, and

(g) using the distributed verification public keys to authenticate executables, including the step of preventing secure execution spaces having different work factor classifications from executing the same executable.

33. A method as in claim 32 further including the step of maintaining the distributed verification public keys within tamper resistant enclosures.

34. A method as in claim 32 further including the step of digitally signing each executable with at least two substantially different, independent techniques and/or by different verifying authorities.

35. A method as in claim 32 further including the step of testing whether the executable satisfies at least one specification at least in part describing the executable operation, and digitally signing the executable and associated specification if the testing step reveals the executable satisfies the specification.

36. A protected processing comprising:

a tamper resistant barrier having a first work factor, and

at least one arrangement within the tamper resistant barrier that prevents the secure execution space from executing the same executable accessed by a further secure execution space having a further tamper resistant barrier with a further work factor substantially different from the first work factor.

37. A secure execution space as in claim 36 wherein the preventing arrangement includes a digital signature authenticating circuit.

38. A secure execution space as in claim 37 wherein the preventing arrangement includes first and second digital signature authenticating circuits applying substantially different digital signature authenticating techniques.

39. A secure execution space as in claim 36 wherein the preventing arrangement comprises means for randomly selecting between first and second, substantially different digital signature authentication techniques.

40. A method for protecting a computation environment surrounded by

a tamper resistant barrier having a first work factor, the method including:

preventing the computation environment from using the same software module accessed by a further computation environment having a further tamper resistant barrier with a further work factor substantially different from the first work factor.

41. A method as in claim 40 wherein the preventing step comprises authenticating at least one digital signature associated with the first-mentioned computation environment as corresponding to the first work factor.

42. A method of protecting computation environments comprising:

(a) associating plural digital signatures with a executable;

(b) authenticating a first subset of the plural digital signatures with a first tamper resistant computation environment; and

(c) authenticating a second subset of the plural digital signatures with a second tamper resistant computation environment different from the first environment.

43. A method as in claim 42 wherein the associating step (a) comprises digitally signing the executable with first and second, different verifying authorities within a web of trust.